BEACON MONITOR EXPERIMENT
Design Document and FSW Problem Statement
version 2.1 10/22/98 10:10 AM

Table of Contents

For information on the Beacon Experiment, please contact:

Rob Sherwood	818-393-5378
Jay Wyatt	818-354-1414
Alan Schlutsmeyer	818-354-4928

1.0 Architecture

With beacon monitor operations, overall spacecraft health is translated into one of four states. Each of these states is represented by a tone that indicates how urgent it is to track the spacecraft for telemetry. These tones will provide assurances to the ground team that the spacecraft is functioning as expected.

In order for beacon monitor operations to be viable, it is necessary to provide ground operators with concise summaries of onboard events since the last contact. The DS-1 technology experiment will consist of several approaches to summarization, one being a capability to produce value-added derived information from telemetry channels.

Telemetry summarizer functions will be made available to the operations team throughout the mission once validated. They can be applied to engineering data channels as needed and have utility outside of the full-on beacon monitor operations concept. The summarization measures target low-bandwidth verification of nominal system behavior across the entire system. They are likely to be especially useful in cross-correlating engineering data when troubleshooting anomalies.

Functionality of the beacon modules will be demonstrated in one or more fault scenarios. It is not necessary that tone selection and engineering data summarization be demonstrated within the same scenario. Plausible uses of summary techniques and tone selection are being sought, but it is more important that the demo show that the software is functionally correct.

In the beacon software, capability exists for the summarization software to provide online comparison of sensor values against functional limits provided by a ground-based trainer. This capability will allow interesting behaviors (such as early signs of anomalies) to be detected and summarized even when they are not detectable from the sets of manually pre-defined limits and rules. The two times at which a sensor first goes outside and then returns within these limits will define additional episodes for the beacon engineering data summarizer. Furthermore, the sensors used as inputs for each of these learned functions will provide a natural and automated criterion for identifying additional sensors that the summarizer should include in its summary of each episode. Ground-based training will be performed shortly after launch on downlinked DS1 data. Eventually, sufficient amounts of available real mission telemetry will allow the trainer to produce refined limit functions that can be uploaded to the spacecraft for more accurate (e.g. more context-appropriate) summarization.

1.1 Overview

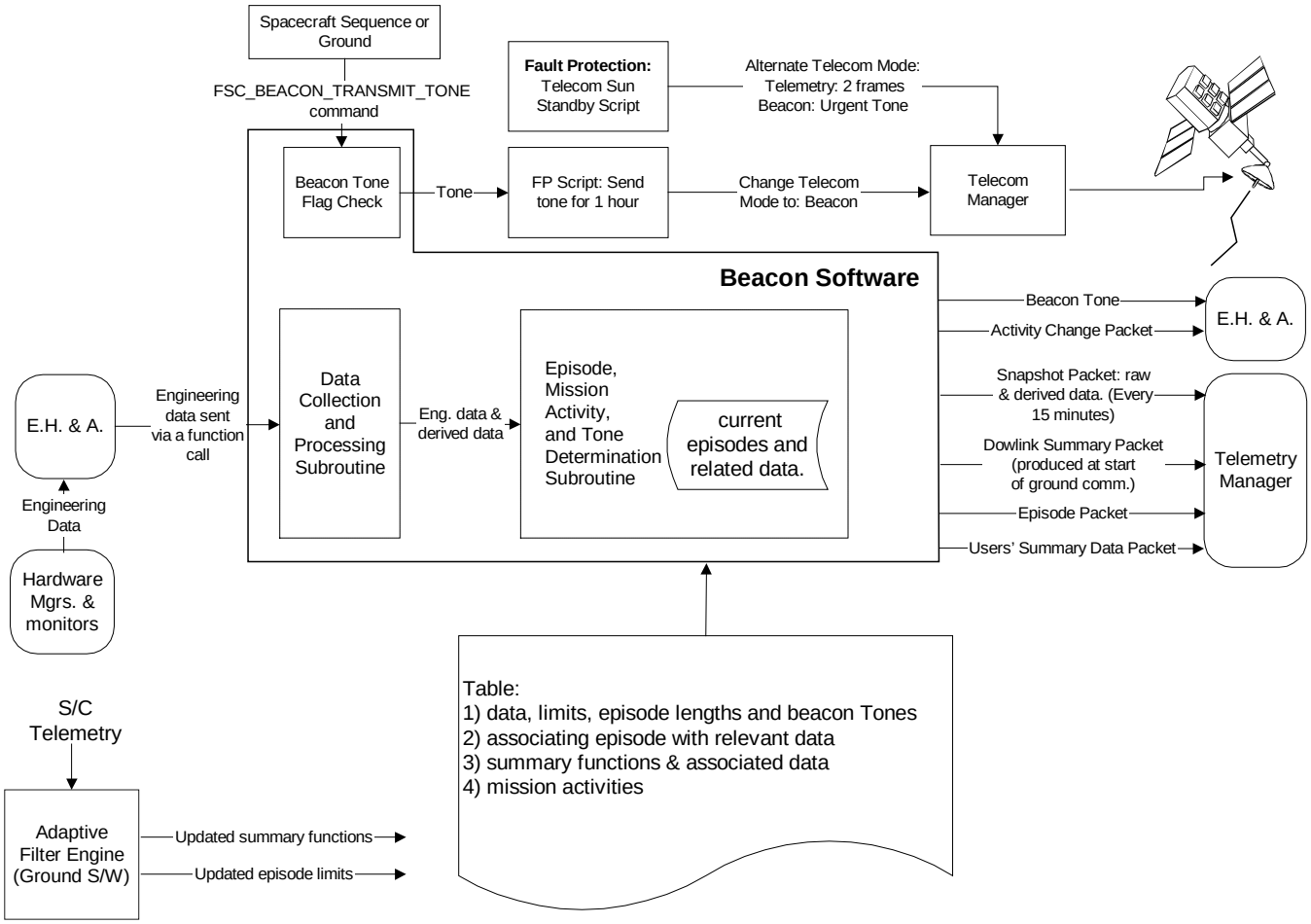
A summary of the main module functions is presented in Table 1:

Software	Functions
Summarizer/ Sampler	1. Receives engineering data (data id, value) via function calls from EHA 2. Computes running summaries of engineering data 3. Stores engineering and summary data internally 4. Uses EHA data to indicate mission activity change 5. Outputs a mission activity telemetry (EHA) packet when activity changes 6. Chooses data alarm limits based on current mission activity 7. Compares engineering and summary data with alarm limits 8. If any limit is exceeded, spawns a new "beacon episode" 9. At end of episode, outputs episode telemetry packet containing details of out of limits conditions, duration, and the measured values of the out of limits data and related data 10. When mission mode changes to DOWNLINK, outputs a top level summary telemetry packet 11. Output data snapshot packets at user-specified intervals (~15 minutes) 12. Outputs user-defined summary telemetry packets at user-specified intervals
Tone Selector	Starting after line 8 from Summarizer/Sampler above: 1. Maps severity of episode to beacon tone (Selects the beacon tone based on severity of fault, interesting or important) 2. Stores current state of beacon; if new state is less severe than previous state, state does not change 3. If beacon tone changes, outputs a telemetry (EHA) packet 4. Resets the beacon tone to NOMINAL upon receipt of command from ground
Scheduled tone transmission	This condition is started by a sequence or ground command: FSC_BEACON_TRANSMIT_TONE() 1. Checks for beacon flag enable, if enabled, continue 2. Check that downlink flag = 0, and XPA is ON, if so, continue (transmitter power amp on, not downlinking) 3. Send the current beacon tone for 1 hour
Fault Protection Standby Response Scripts	The fault protection standby response scripts will put the spacecraft in a safe state and start sending emergency telemetry using the low gain antenna. With the beacon flag enabled, the scripts will alternate between sending emergency telemetry (for 30 minutes) and the beacon tone (for 25 minutes.) The 30 minute time is sufficient to receive 2 packets of telemetry data. The 25 time is sufficient to decode the beacon tone and the furthest expected spacecraft distance.

Table 1

1.2 Flight Software Flow Diagram

The following diagram shows the data flow through the beacon software.



2.0 Beacon Tone Selector Module

The tone selector module maps EHA engineering data to beacon tone states. The output of this module is the tone state as an EHA telemetry. There are two different operations that will cause the tone to be transmitted by the DS1 Small Deep Space Transponder (SDST). The two operations are fault protection standby mode and a command (sequence or ground.) Both operations require the beacon flag to be enabled. The beacon flag is a binary software flag stored in EEPROM to survive power resets of the on-board CPU. The flag can be commanded from the ground. The four beacon tones are defined in Table 2.

Tone:	Definition:
NOMINAL	Spacecraft is nominal, all functions are performing as expected. No need to down link engineering telemetry.
INTERESTING	An interesting but not urgent event has occurred on the spacecraft. Establish communication with the ground if convenient to obtain data on the interesting event. Example: device reset to clear error caused by SEU or other transient event.
IMPORTANT	The spacecraft needs servicing. Communication with the ground needs to be achieved within a certain time or the spacecraft state could deteriorate and/or critical data could be lost. This time could be a function of the following: time until the next downlink, the time that without the device the S/C attitude will deteriorate to an emergency state, etc. Examples: solid state memory near full, non-critical technology hardware failure.
URGENT	Spacecraft emergency. A critical component of the spacecraft has failed. The spacecraft cannot autonomously recover and ground intervention is required immediately. Examples: 1553 bus failure, PDU failure, SRU failure, IPS gimbal stuck.
NO TONE	Beacon mode is not operating, spacecraft telecom is not Earth-pointed or spacecraft anomaly prohibited tone from being sent.

Table 2

2.1 Tone Selector Inputs

The beacon tone selector uses data from the beacon sampler module to set the beacon tones. The beacon sampler receives data from the EHA subsystem via a function call which is described in the sampler/summarization section.

2.2 Tone Selector Outputs

The EHA data are received by the sampler module and checked against a table of limits. If any of the high/low limits in the table are violated, the beacon tone selector software will output messages to the telemetry manager and the EHA manager.

A telemetry packet is formed and sent to the telemetry manager using the FSW provided macro capabilities. The description of this packet and all telemetry output is in section 3.2. The mission activity is sent in the same packet.

2.3 Tone Selector Commanding

The following command enables/disables the beacon flag. The beacon flag must be enabled in order to transmit the beacon tone. The tone is actually transmitted using an on-board command sequence that sets up the telecom system to send the tone. (See BMOX_TONE_XMIT command and Appendix A.) The urgent tone can also be transmitted by the fault protection system in standby mode. (See Appendix B.) The command has the form:

BMOX_TONE_STATE(0-2) (0=OFF, 1=ON, 2=RESET)

This command sends the following IPC message to the BMOX S/W:
FSC_BEACON_SET_TONE_SELECTOR_STATE (FSC_BEACON_STATE_xxx)
where xxx=

ON = Turns on the output of beacon tone selector defined above. Also, outputs a FSC_BEACON_CURRENT_TONE message.

OFF = Turns off all beacon mode output. The beacon software will no longer listen to FAULT_STATE component of the MIR_STATE_UPDATE messages.

RESET = Resets the last known state to NOMINAL and outputs a NOMINAL tone message and telemetry.

The following command will start a on-board sequence to turn on the beacon tone that corresponds to the state recorded in the BMOX tonesel task. The following conditions must be met for this command to succeed:

- Beacon tone flag is enabled
- XPA power is on
- No downlink is in progress

It will be the responsibility of the tone sequence to turn the tone back off again after a preprogrammed time interval; thus, there is no need for another command to turn the tone off.

BMOX_TONE_XMIT()

This command sends the following IPC message to the BMOX S/W:
FSC_BEACON_TRANSMIT_TONE()

The following command changes the current beacon tone state to the commanded TONE. It may be used to test setting the beacon tone manually. The tone can only be changed to a higher state. This command does not transmit the beacon tone.

BMOX_TONE_VAL(TONE)

This command sends the following IPC message to the BMOX S/W:
FSC_BEACON_CURRENT_TONE_SELECT (TONE)

3.0 Data Summarizer/Sampler Module

The summarizer/sampler module combines three functions: data collection and processing, mission activity determination, and episode identification. The data collection subroutine receives engineering data from the EHA system via a function call and applies summary techniques to these data, producing summary measures for downlink to the ground.

The mission activity subroutine determines the overall spacecraft mode of operation. This determination is used to choose the appropriate data and limits for a particular episode in the episode subroutine. For example, it is logical to assume that the limits for an episode during IPS thrusting would vary from the limits during a maneuver. The mission activity is intended to be exclusive. When a new mission activity starts, the previous mission activity is assumed to have ended.

The episode subroutine determines what data is relevant to each particular episode. The episode subroutine combines summary and engineering data received internally from the summarizer/sampler module with the mission activity received from the activity subroutine and compares the data with mission activity specific alarm limits. Some of these limits are computed with an adaptive filtering technology called ELMER.

It is important to only consider data that is relevant to the current mission activity. For example, IPS sensor values may be important while using IPS, but if the satellite is in RCS control mode then IPS sensor values could be ignored. In addition, the ACS rate limits might be different during cruise than during a maneuver. As these examples point out, it is necessary to use the mission activities to determine which data to use for episode identification and what are the limits of these data. If the limit is exceeded, the subroutine spawns a new episode and collects past relevant data from summarizer/sampler. The past data collected will be one-minute summaries that go back (episode length) minutes from start of episode. (So a five-minute episode would contain summaries starting five minutes before the episode to five minutes after the episode.) At the end of the episode, the subroutine outputs episode name (out of limit data ID), high limit, low limit, relevant data, start and end times (relevant data is a vector of the one minute summaries.)

3.1 Data Summarizer/Sampler Inputs

Data Collection and Processing Inputs

The data collection and processing subroutine is a simple data collection service which uses a C function to send data items from EHA to Beacon's data pool:

extern int initBeaconIdTable(int *Object_ID_List, int N_Meas_Objects);

which is called at initialization time to set up the table of EHA values of interest to Beacon.

extern void getBeaconMeasurements(FSC_BEACON_data_value_t *rack);

which is called once each processing cycle to retrieve the set of EHA measurements needed by Beacon. This set is defined in the samplerInitFile and is listed in Table 3.

Once the data are received, the following functions are applied to the data as necessary:

- Running Maximum
- Running Minimum
- Running Average
- 1st derivative
- 2nd derivative

In addition, the ground trainer software may develop combinations of the above functions for data summarization and episode identification.

EH&A Table ID	Description
ACS_TELEM_ALLOCATED_ENTRY_2	ACS SAC controller error (SADA ang. error in x-axis)
ACS_TELEM_ALLOCATED_ENTRY_8	ACS ATE s/c rate estimate element (rate ω_x in rad/s)
ACS_TELEM_ALLOCATED_ENTRY_28	ACS SAC controller error (SADA ang. error in y-axis)
ACS_TELEM_ALLOCATED_ENTRY_32	ACS ATE s/c rate estimate element (rate ω_y in rad/s)
ACS_TELEM_ALLOCATED_ENTRY_51	ACS ATE s/c rate estimate element (rate ω_z in rad/s)
ACS_TELEM_ALLOCATED_ENTRY_182	Current error rate ω_x in rads/s
ACS_TELEM_ALLOCATED_ENTRY_184	Current error x-axis rads
ACS_TELEM_ALLOCATED_ENTRY_188	Current error rate ω_y in rads/s
ACS_TELEM_ALLOCATED_ENTRY_191	Current error y-axis rads
ACS_TELEM_ALLOCATED_ENTRY_194	Current error rate ω_z in rads/s
ACS_TELEM_ALLOCATED_ENTRY_214	Current error z-axis rads
MON_CONTROL_ERROR_MON_EHA_F_X	X axis control error
MON_CONTROL_ERROR_MON_EHA_F_Y	Y axis control error
MON_CONTROL_ERROR_MON_EHA_F_Z	Z axis control error
FSC_BATTERY_1_SOC	Battery 1 state of charge
FSC_BATTERY_2_SOC	Battery 2 state of charge
FSC_BATTERY_MID_VOLT_1_MEAS	Battery 1 midpoint voltage
FSC_BATTERY1_CURRENT_MEAS	Battery 1 current
FSC_BATTERY_MID_VOLT_2_MEAS	Battery 2 midpoint voltage
FSC_BATTERY2_CURRENT_MEAS	Battery 2 current
FSC_SCARLET_VOLT_MEAS	Solar Array voltage
FSC_SCARLET_WING1_CUR_MEAS	Solar Array 1 current
FSC_SCARLET_WING2_CUR_MEAS	Solar Array 2 current
FSC_PDU_ESS_BUS_CUR_MEAS	Essential Bus current
FSC_PDU_ESS_BUS_VOL_MEAS	Essential Bus voltage at PDU
FSC_PDU_NEB1_CUR_MEAS	Non-Essential Bus 1 current
FSC_PDU_NEB2_CUR_MEAS	Non-Essential Bus 2 current
FSC_PDU_NEB3_CUR_MEAS	Non-Essential Bus 3 current
FSC_PDU_RELAY_FET_STATUS_WORD0_MEAS	Digital channel
FSC_PDU_RELAY_FET_STATUS_WORD1_MEAS	Digital channel
FSC_PDU_RELAY_FET_STATUS_WORD2_MEAS	Digital channel
FSC_SDST_XPDR_STATE_MEAS	Digital channel
FSC_SDST_X_PWR_MEAS	Digital channel
FSC_SDST_EXCITER_SPE_MEAS	Digital channel
FSC_PDU_SDST_CUR_MEAS	SDST Current
FSC_PDU_KASSPA_CUR_MEAS	XPA Power Status
FSC_PDU_XSSPA_CUR_MEAS	X-SSPA current

MON_ACS_INFO_EHA_MDC_STATE	ACS state
DWN_PRYOR_STATE_0	Downlink priority table state

Table 3

The digital channels listed near the end of Table 3 are further derived into other channels which can be summarized by the beacon software. The names of the derived channels are listed in Table 4. The source channel from which they are derived is also listed.

Name (new channel)	Source Channel	# bits	Start bit	lag
XPDR_STATE_X	FSC_SDST_XPDR_STATE_MEAS	4	1	60
XPDR_STATE_KA	FSC_SDST_XPDR_STATE_MEAS	5	1	60
XPA_INPUT_POWER	FSC_SDST_X_PWR_MEAS	8	8	60
EXCITER_SPE_KA	FSC_SDST_EXCITER_SPE_MEAS	0	8	60
EXCITER_SPE_X	FSC_SDST_EXCITER_SPE_MEAS	8	8	60
XPA_POWER_ON	FSC_PDU_RELAY_FET_STATUS_WORD0_MEAS	11	1	60

Table 4

Mission Activity Subroutine Inputs

The mission activity subroutine monitors particular EHA engineering telemetry that indicate a mission mode change. When the mission mode changes, the episode subroutine is reset and new limits are loaded. A telemetry message is generated at the same time. Table 5 contains the mission activity list. When the mission activity changes to RCS with downlink, the software outputs a downlink summary packet. The EHA data channel that is used to determine the mission activity is the MON_ACS_EHA_MDC_STATE. This channel has several enumerated types that determine the spacecraft ACS state. In addition, the DWN_PRYOR_STATE_0 data channel is necessary to determine when the spacecraft is downlinking. This channel gives the status of the "no-downlink" downlink priority table. When this data channel is off, one of the other downlink priority table channels must be on and therefore, the spacecraft must be downlinking. These channels are "hard-coded" into the beacon software. Any change of these channels in EHA would require a software change from Beacon. This is not anticipated during the DS1 mission because these are critical channels.

Mission Activity	**EHA** (MON_ACS_EHA_MDC_STATE) **State**
Downlink	MON_ACS_RCS_MODE and downlink priority flag 0 set to off
Cruise (no downlink)	MON_ACS_RCS_MODE
Maneuver	MON_ACS_RCS_DV_MODE
IPS thrusting	MON_ACS_TVC_MODE
Standby mode	MON_ACS_SUN_STANDBY_SSA or MON_ACS_SUN_STANDBY_SRU or MON_ACS_EARTH_STANDBY_SSA or MON_ACS_EARTH_STANDBY_SRU

Table 5

Episode Subroutine Inputs

The following four tables (Table 6 - Table 9) contain the data alarm limits for each of the four mission activities. The limits in these tables are stored in the samplerInitFile.

Mission Activity: Cruise																
Data ID	**Min./Max.**				**Average**				**1st deriv.**				**2nd deriv.**			
	Min.	**tone**	**max.**	**tone**	**min.**	**tone**	**max.**	**tone**	**min.**	**tone**	**max.**	**tone**	**min.**	**tone**	**max.**	**tone**
FSC_SCARLET_VOLT_MEAS	85	IN	135	IM												
FSC_BATTERY_1_SOC	75	IN							-3	IM	3	IM				
FSC_BATTERY_1_SOC	40	IM	105	IN												
FSC_BATTERY_2_SOC	75	IN							-3	IM	3	IM				
FSC_BATTERY_2_SOC	40	IM	105	IN												
FSC_PDU_ESS_BUS_CUR_MEAS	10	IN	99	IM												
FSC_PDU_ESS_BUS_VOL_MEAS	24	IM	34	IM												
FSC_PDU_SDST_CUR_MEAS			3	IN												
FSC_PDU_XSSPA_CUR_MEAS			3	IN												
FSC_PDU_NEB1_CUR_MEAS			90	IN												
FSC_PDU_NEB2_CUR_MEAS			90	IN												
FSC_PDU_NEB3_CUR_MEAS			90	IN												
FSC_PDU_KASSPA_CUR_MEAS			20	IN												

Table 6

Mission Activity: Downlink																
Data ID	**Min./Max.**				**Average**				**1st deriv.**				**2nd deriv.**			
	Min.	**tone**	**max.**	**tone**	**min.**	**tone**	**max.**	**tone**	**min.**	**tone**	**max.**	**tone**	**min.**	**tone**	**max.**	**tone**
FSC_SCARLET_VOLT_MEAS	85	IN	135	IM												
FSC_BATTERY_1_SOC	65	IN							-3	IM	3	IM				
FSC_BATTERY_1_SOC	40	IM	105	IN												
FSC_BATTERY_2_SOC	65	IN							-3	IM	3	IM				
FSC_BATTERY_2_SOC	40	IM	105	IN												
FSC_PDU_ESS_BUS_CUR_MEAS	24	IN	34	IM												
FSC_PDU_ESS_BUS_VOL_MEAS	75	IM	99	IM												
FSC_PDU_SDST_CUR_MEAS	10.8	IN	14.5	IM												
FSC_PDU_XSSPA_CUR_MEAS	38.5	IN	72.0	IM												
FSC_PDU_NEB1_CUR_MEAS			90	IN												
FSC_PDU_NEB2_CUR_MEAS			90	IN												
FSC_PDU_NEB3_CUR_MEAS			90	IN												
FSC_PDU_KASSPA_CUR_MEAS			20	IN												

Table 7

Mission Activity: Maneuver																
Data ID	**Min./Max.**				**Average**				**1st deriv.**				**2nd deriv.**			
	Min.	**tone**	**max.**	**tone**	**min.**	**tone**	**max.**	**tone**	**min.**	**tone**	**max.**	**tone**	**min.**	**tone**	**max.**	**tone**
FSC_SCARLET_VOLT_MEAS	85	IN	135	IM												
FSC_BATTERY_1_SOC	65	IN							-3	IM	3	IM				
FSC_BATTERY_1_SOC	40	IM	105	IN												
FSC_BATTERY_2_SOC	65	IN							-3	IM	3	IM				
FSC_BATTERY_2_SOC	40	IM	105	IN												
FSC_PDU_ESS_BUS_CUR_MEAS	0	IN	99	IM												
FSC_PDU_ESS_BUS_VOL_MEAS	22	IM	34	IM												
FSC_PDU_SDST_CUR_MEAS	10.8	IN	14.5	IM												
FSC_PDU_XSSPA_CUR_MEAS	38.5	IN	72.0	IM												
FSC_PDU_NEB1_CUR_MEAS			90	IN												
FSC_PDU_NEB2_CUR_MEAS			90	IN												
FSC_PDU_NEB3_CUR_MEAS			90	IN												
FSC_PDU_KASSPA_CUR_MEAS			20	IN												

Table 8

Mission Activity: IPS Thrusting																
Data ID	**Min./Max.**				**Average**				**1st deriv.**				**2nd deriv.**			
	min.	**tone**	**max.**	**tone**	**min.**	**tone**	**max.**	**tone**	**min.**	**tone**	**max.**	**tone**	**min.**	**tone**	**max.**	**tone**
FSC_SCARLET_VOLT_MEAS	85	IN	135	IM												
FSC_BATTERY_1_SOC	65	IN							-3	IM	3	IM				
FSC_BATTERY_1_SOC	40	IM	105	IN												
FSC_BATTERY_2_SOC	65	IN							-3	IM	3	IM				
FSC_BATTERY_2_SOC	40	IM	105	IN												
FSC_PDU_ESS_BUS_CUR_MEAS	0	IN	99	IM												
FSC_PDU_ESS_BUS_VOL_MEAS	22	IM	34	IM												
FSC_PDU_SDST_CUR_MEAS	10.8	IN	14.5	IM												
FSC_PDU_XSSPA_CUR_MEAS	38.5	IN	72.0	IM												
FSC_PDU_NEB1_CUR_MEAS			90	IN												
FSC_PDU_NEB2_CUR_MEAS			90	IN												
FSC_PDU_NEB3_CUR_MEAS			90	IN												
FSC_PDU_KASSPA_CUR_MEAS			20	IN												

Table 9

3.2 Data Summarizer/Sampler Outputs

The telemetry packet containing the mission activity is the beacon_state packet. It was previously described in Section 2.2 Tone Selector Outputs. The other summarizer/sampler telemetry is output in three telemetry messages. The first two messages listed has several sub-types.

BEACON_stats (statistics since the previous downlink)

NAME	BITS	FORMAT	DESCRIPTION
type	8	int	type of event (each type has its own format), 0 = downlink header, 1 = channel summary

This data packet has two different types which are defined below.

type 0: (downlink header - one per downlink)

end time	32	int	time this reporting period ended (sclk)
end ephem time	32	int	ephemeris time this reporting period ended
duration	32	int	# of seconds in this report
channel count	16	int	# of channel summary records to follow

type 1: (channel summary - one per channel per downlink -- only produced for channels which were out-of-limits since the previous downlink)

channel	16	int	channel id
episode count	16	int	# of episodes this channel was out-of-limits
worst minimum	32	flt	worst error below the minimum limit
worst maximum	32	flt	worst error above the maximum limit

BEACON_event (data pertaining to detected or scheduled events)

NAME	BITS	FORMAT	DESCRIPTION
type	32	int	type of event (each type has its own format), 0 = tone change, 1 = activity change, 2 = sample interval change, 3 = episode summary, 4 = episode channel, 5 = user summary

This data packet has six different types which are defined below.

type 0: (tone change - produced when remembered tone changes)

tone	32	enum	currently selected Beacon tone: 0 = NOMINAL, 1 = INTERESTING, 2 = IMPORTANT, 3 = URGENT

type 1: (activity change - produced when mission activity code changes)

activity	32	enum	current spacecraft mission activity code 0 = DOWNLINK, 1 = CRUISE, 2 = IPS, 3 = MANEUVER, 4 = STANDBY

type 2: (sampler interval change - produced when SET_SAMPLE_INTERVALS command is received)

sample interval	16	int	`# of seconds between samples`

save interval	16	int	# of sample intervals between saved samples
hist. interval	16	int	# of save intervals between snapshots

type 3: episode summary - one per episode

onset time	32	int	time the episode began (sclk)
history time	32	int	time of last history value in channel packets (sclk)
duration	16	int	# of seconds in this episode
channels out	16	int	# of channels out-of-limits during episode
related chan.	16	int	# of related channels reported

type 4: episode channel - one per reported channel in each episode

channel	16	int	channel id
out count	16	int	# of samples channel was out-of-limits
low limit	32	flt	channel low limit in effect during episode
high limit	32	flt	channel high limit in effect during episode
onset value	32	flt	channel value at onset of episode
minimum value	32	flt	channel minimum value during episode
maximum value	32	flt	channel maximum value during episode
mean value	32	flt	channel mean value during episode
history count	16	int	# of values in the history table
history table	32	flt	per value - value of the channel at one minute intervals starting up to 5 minutes before onset of episode, ending at end of episode

type 5: user summary - contents defined by user, packet produced at a regular interval also specified by user

start time	32	int	start time of summary period (sclk)
end time	32	int	end time of summary period (sclk)
sample count	32	int	number of samples taken per value
generation	32	int	generation # of associated samplerInitFile
packet id	16	int	user packet identifier code
value count	16	int	number of values in the value table
value table	32	flt	per value - user defined values

BEACON_data_sample (a snapshot of values, 1 for each defined channel, taken at the end of each history interval)

NAME	BITS	FORMAT	DESCRIPTION
generation	32	int	generation # of associated samplerInitFile
sample time	32	int	time sample was taken (sclk)
sample data	32	flt	per value - channel values (up to 250 values max)

3.3 Data Summarizer/Sampler Commanding

The output of the module can be turned OFF, ON or RESET by issuing a ground command. The command has the form:

BMOX_SUM_STATE(OFF, ON, or RESET)

Internally, this command will issue an IPC message to the beacon software with the following syntax:

FSC_BEACON_SET_SUM_STATE(FSC_BEACON_STATE_xxx)

where xxx = ON or OFF

BMOX_SUM_INTERVL(seconds, sample, summary)

This command specifies the three intervals used by the summarizer. The first is the number of seconds between raw data samples. The second is the interval between saved summaries (used for the episode history values). The third is the interval between BEACON_data_sample packets. For example, BMOX_SUM_INTERVL(5,6,16) will cause Sampler to:

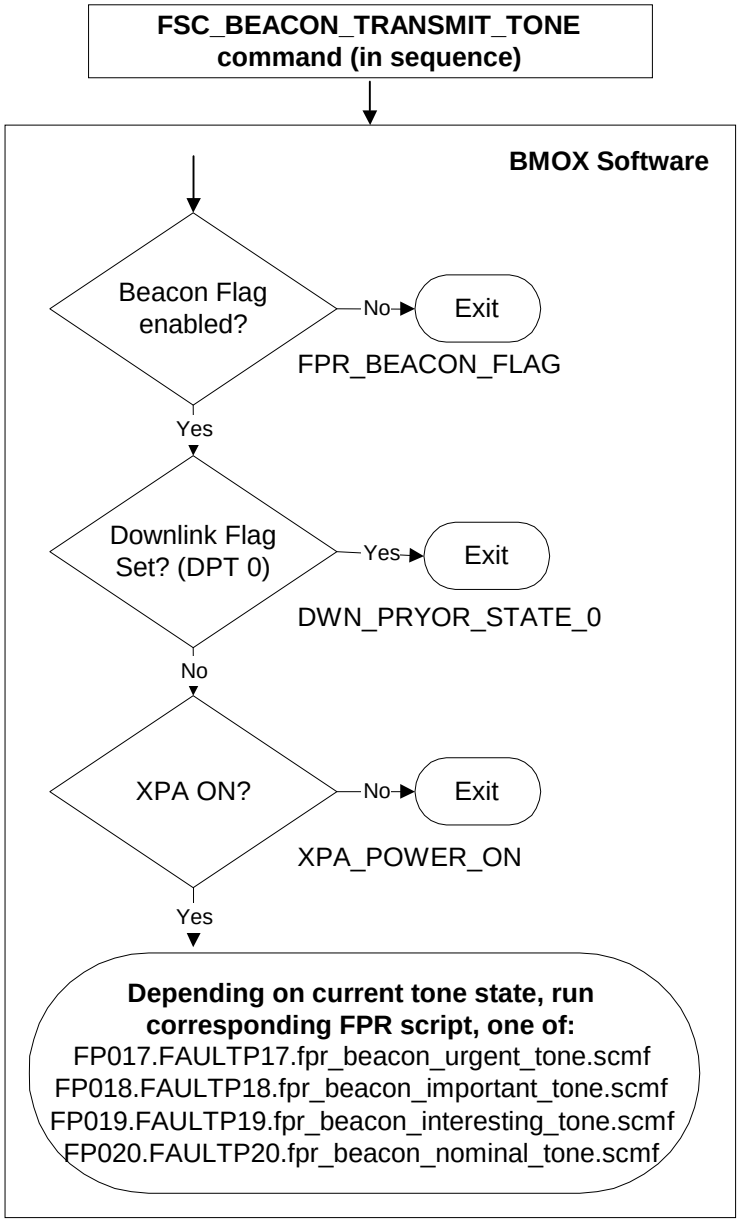
- Take one raw data sample every 5 seconds
- Save one summary sample every 30 seconds (every 6 raw samples)
- Produce one BEACON_data_sample telemetry packet every eight minutes (30x16 = 480 seconds)

The default setting for these values is BMOX_SUM_INTERVL(1,60,15), or one raw sample per second, one summary sample every minute and one history sample every 15 minutes. Internally, this command will issue an IPC message to the beacon software with the following syntax:

FSC_BEACON_SET_SUM_INTERVALS(1,60,15)

Changing the intervals will reset the summarization software.

Appendix A: Scheduled Beacon Tone Transmission

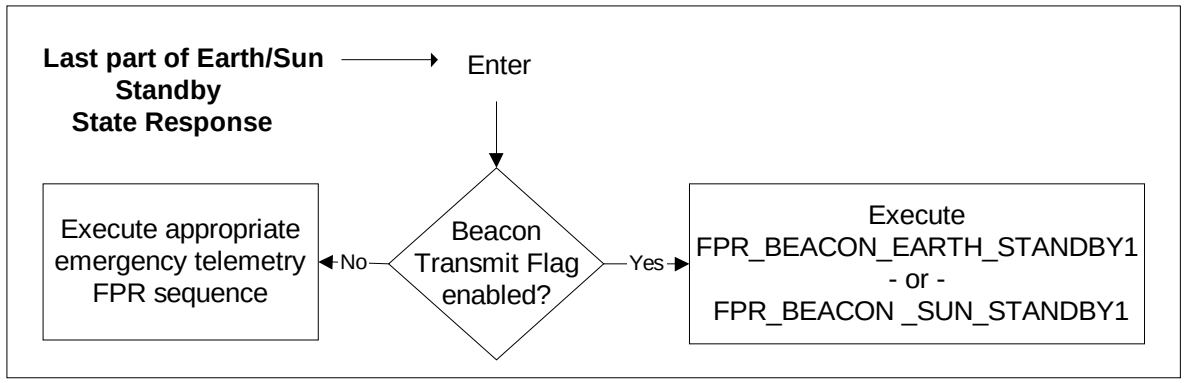


FP0xx.FAULTPxx.fpr_beacon_ *tonename*_tone.scmf

```
COHERENCY                 00:00:01.000,0010025897:2,VC2,"DISABL"
COHERENCY                 00:00:02.000,0010025897:2,VC2,"DISABL"
SET_ENCODING    00:00:03.000,0010025905:2,VC2,"X","OFF"
SET_ENCODING    00:00:04.000,0010025905:2,VC2,"X","OFF"
SET_DL_MOD_INDEX          00:00:05.000,0010025901:2,VC2,"X",48
SET_DL_MOD_INDEX          00:00:06.000,0010025901:2,VC2,"X",48
SET_SUBCR_FREQ            00:00:07.000,0010025900:2,VC2,"X","toneval"
SET_SUBCR_FREQ            00:00:08.000,0010025900:2,VC2,"X","toneval"
EXCITER_POWER 00:00:09.000,0010025917:2,VC2,"X","ON"
EXCITER_POWER 00:00:10.000,0010025917:2,VC2,"X","ON"
(turn off stuff next)
SET_ENCODING    01:00:11.000,0010025905:2,VC2,"X","15_6"
SET_ENCODING    01:00:12.000,0010025905:2,VC2,"X","15_6"
EXCITER_POWER 01:00:13.000,0010025917:2,VC2,"X","OFF"
EXCITER_POWER 01:00:14.000,0010025917:2,VC2,"X","OFF"
```

where toneval is:
20 Urgent tone
25 Important tone
30 Interesting tone
35 Nominal tone

Appendix B: Standby Mode Beacon Tone Transmission



The standby scripts that are run at the end of this tree are listed on the next two pages.

FPR_BEACON_EARTH_STANDBY1

0:00:00 SDST_RESET Reset to ensure good configuration
0:00:01 SDST_RT_TIMEOUT ENABLE Required after reset
0:00:02 SET_UPLINK 7.8125 7.8125 bps
0:00:03 COHERENCY DISABL coherency mode disabled (TWNC on)
0:00:04 RANGING X OFF x-band ranging off
0:00:05 SET_SUBCR_MODE X SUBCR x-band sub-carrier x-band sub carrier
0:00:06 SET_SUBCR_FREQ X 375 x-band sub-carrier frequency to 375 kHz through ~L+285 days, then change to 25
0:00:07 SET_DL_MOD_INDEX X 42 modulation index for downlink to 72°
0:00:08 SET_ENCODING X 15_6 x-band encoding mode to 15, 1/6
0:00:09 RANGING KA OFF Ka-band ranging off
0:00:10 SET_SUBCR_MODE KA SUBCR Ka-band subcarrier on
0:00:11 SET_SUBCR_FREQ KA 375 Subcarrier frequency 375 kHz through ~L+285 days, then change to 25
0:00:12 SET_DL_MOD_INDEX KA 45 modulation index for downlink to 68°
0:00:13 SET_ENCODING KA 15_6 select (15,1/6) convolutional encoder
0:00:14 SET_DOWNLINK 2100 downlink rate to 2100 bps, later to 300 bps
0:00:15 SET_DL_CLK_RATE 6X clk rate to 6X
0:00:16 RS_ENCODING ON Reed-Solomon encoding enabled
0:00:17 EXCITER_POWER KA OFF ka-band exciter off
0:00:18 EXCITER_POWER X OFF x-band exciter off
0:00:19 SELECT_ANT_GAIN HGA WTS1 to HGA
0:00:20 XSSPA_POWER ON xpa on
0:00:21 KASSPA_POWER ON kapa on
0:00:27 EXCITER_POWER X ON x-band exciter on
0:00:33 EXCITER_POWER KA ON ka-band exciter on
0:00:43 FP_FAULT_OP XPA ENABLE enable xpa fault
0:00:44 FP_FAULT_OP SDST ENABLE enable sdst fault
0:00:45 FP_MON_OP DWNLNK SC_ENA enable downlink monitor
0:03:45 SET_EHA_RATE 5,5,4,4 rt & rec = 5 sec, r/t & rec update = 4
0:33:45 ACTIVAT_SEQUENCE FAULTP 22 activate beacon earth standby2

FPR_BEACON_SUN_STANDBY1

0:00:00 SDST_RESET Reset to ensure good configuration
0:00:01 SDST_RT_TIMEOUT ENABLE Required after reset
0:00:02 SET_UPLINK 7.8125 7.8125 bps
0:00:03 COHERENCY DISABL coherency mode disabled (TWNC on)
0:00:04 RANGING X OFF x-band ranging off
0:00:05 SET_SUBCR_MODE X SUBCR x-band sub-carrier x-band sub carrier
0:00:06 SET_SUBCR_FREQ X 25 x-band sub-carrier frequency to 25 kHz
0:00:07 SET_DL_MOD_INDEX X 40 modulation index for downlink to 69°
0:00:08 SET_ENCODING X 7_2 x-band encoding mode to 7 1/2
0:00:09 SET_DOWNLINK 2100 2100 bps, then 40 bps, then 10 bps
0:00:10 SET_DL_CLK_RATE 2X clk rate to 2X
0:00:11 RS_ENCODING ON Reed-Solomon encoding enabled
0:00:12 EXCITER_POWER KA OFF ka-band exciter off
0:00:13 EXCITER_POWER X OFF x-band exciter off
0:00:14 SELECT_ANT_GAIN LGA WTS1 to LGA
0:00:15 SELECT_LGA X select LGA +X
0:00:16 KASSPA_POWER OFF kapa off
0:00:17 XSSPA_POWER ON xpa on
0:00:23 EXCITER_POWER X ON x-band exciter on
0:00:33 FP_FAULT_OP XPA ENABLE enable xpa fault
0:00:34 FP_FAULT_OP SDST ENABLE enable sdst fault
0:00:35 FP_MON_OP DWNLNK SC_ENA enable downlink monitor
0:03:35 SET_EHA_RATE 5,5,4,4 rt & rec = 5 sec, r/t & rec update = 4
0:33:35 ACTIVAT_SEQUENCE FAULTP 21 activate beacon sun standby2

These scripts set up telecom for emergency telemetry and then run the beacon script on the next page. They are similar to the standard fault protection telecom scripts except they run the beacon script at the end. There are scripts for both Earth standby and sun standby. The result of this series of scripts is to toggle between emergency telemetry (30 minutes) and beacon tone (25 minutes).

FPR_BEACON_SUN_STANDBY2

0:00:01 SET_ENCODING X OFF x-band encoding mode off
0:00:02 SET_ENCODING X OFF x-band encoding mode off
0:00:03 SET_SUBCR_FREQ X 20 x-band sub-carrier frequency to 20 kHz
0:00:04 SET_SUBCR_FREQ X 20 x-band sub-carrier frequency to 20 kHz
0:00:05 SET_DL_MOD_INDEX X 48 modulation index
0:00:06 SET_DL_MOD_INDEX X 48 modulation index
0:00:07 BMOX_TONE_VAL 3 urgent tone
0:25:07 ACTIVAT_SEQUENCE FAULTP 23 activate beacon sun standby3

FPR_BEACON_SUN_STANDBY3

0:00:01 SET_ENCODING X 7_2 x-band encoding mode to 7 1/2
0:00:02 SET_ENCODING X 7_2 x-band encoding mode to 7 1/2
0:00:03 SET_SUBCR_FREQ X 25 x-band sub-carrier frequency to 25 kHz
0:00:04 SET_SUBCR_FREQ X 25 x-band sub-carrier frequency to 25 kHz
0:00:05 SET_DL_MOD_INDEX X 40 modulation index for downlink to 69°
0:00:06 SET_DL_MOD_INDEX X 40 modulation index for downlink to 69°
0:30:06 ACTIVAT_SEQUENCE FAULTP 21 activate beacon sun standby2

FPR_BEACON_EARTH_STANDBY2

0:00:01 SET_ENCODING X OFF x-band encoding mode off
0:00:02 SET_ENCODING X OFF x-band encoding mode off
0:00:03 SET_SUBCR_FREQ X 20 x-band sub-carrier frequency to 20 kHz
0:00:04 SET_SUBCR_FREQ X 20 x-band sub-carrier frequency to 20 kHz
0:00:05 SET_DL_MOD_INDEX X 48 modulation index
0:00:06 SET_DL_MOD_INDEX X 48 modulation index
0:00:07 BMOX_TONE_VAL 3 urgent tone
0:25:07 ACTIVAT_SEQUENCE FAULTP 24 activate beacon earth standby3

FPR_BEACON_EARTH_STANDBY3

0:00:01 SET_ENCODING X 15_6 select (15,1/6) convolutional encoder
0:00:02 SET_ENCODING X 15_6 select (15,1/6) convolutional encoder
0:00:03 SET_SUBCR_FREQ X 375 Subcarrier frequency 375 kHz through ~L+285 days, then change to 25
0:00:04 SET_SUBCR_FREQ X 375 Subcarrier frequency 375 kHz through ~L+285 days, then change to 25
0:00:05 SET_DL_MOD_INDEX X 42 modulation index for downlink to 72°
0:00:06 SET_DL_MOD_INDEX X 42 modulation index for downlink to 72°
0:30:06 ACTIVAT_SEQUENCE FAULTP 22 activate beacon earth standby2

These scripts continues to send emergency telemetry for 30 minutes. They then reconfigure for beacon tone transmission and send a tone for 25 minutes. After 25 minutes of tone, they run the last script (standby3).

These scripts reconfigure for emergency telemetry and loop back to the standby2 script.